第32卷 第13期 计 算 机 工 程 2006年7月
Vol.32 № 13 Computer Engineering July 2006
·软件技术与数据库· 文章编号：1000—3428(2006)13—0035—03 文献标识码：A 中图分类号：TP311.13

全局数据目录的动态管理和维护

陈建英[1,2]，刘心松[2]，谈文蓉[1]，刘 韬[1]，谭 颖[1]，王 莉[1]
(1. 西南民族大学计算机科学与技术学院，成都 610041；2. 电子科技大学 8010 研究室，成都 610054)

摘 要：针对分布式并行数据库系统DPDBS中数据库副本数、数据库分布等信息的动态性，提出全局数据目录的动态管理和维护算法。该算法将数据目录一致性检查嵌入数据库执行过程中，通过改进两阶段提交协议和采用捎带、恢复和并行处理等技术，在低开销的情况下动态地保证了全局数据信息的一致性和实时性。

关键词：分布式并行数据库系统；全局数据目录；两阶段提交协议；一致性

Dynamic Management and Maintenance of Whole Data Directory

CHEN Jianying[1,2], LIU Xinsong[2], TAN Wenrong[1], LIU Tao[1], TAN Ying[1], WANG Li[1]
(1. School of Computer Science & Technology, Southwest University for Nationalities, Chengdu 610041;
2. 8010 Research and Development Group, University of Electronic Science & Technology of China, Chengdu 610054)

【Abstract】 Aiming at the dynamic characteristic of the data directory in distributed and parallel database system (DPDBS), the dynamic arithmetic for whole data directory is put forward. The arithmetic is implemented by embedding the consistency check of whole data directory in the course of database executing, improving the two phase commit protocol, and making use of technology such as recovery, parallel handling, and other technology. The application proves its efficiency.

【Key words】 Distributed and parallel database system; Whole data directory; Two phase commit protocols(2PC); Consistency

1 分布式并行数据库系统和数据目录概述

分布式并行数据库系统(DPDBS)的体系结构以及设计特点克服了以往分布式并行系统中服务器和客户交互只由一个通道完成的瓶颈[1,2]。但是，多通道的存在使全局数据一致性的管理和维护远比集中式管理系统复杂，不仅要求数据库信息与物理信息的一致性，还要求全局数据库信息本身的一致性。在DPDBS中，全局数据库信息存放在一个称为“数据目录”的全局大粒度数据字典中，包含数据库名、数据库存放位置以及数据库状态等基本信息，为保证全局数据一致性，就要求这些信息能够随着系统中节点的加入和退出、数据库副本数的增加和减少及数据库存放位置的改变而动态改变。

数据目录中的数据库信息与物理信息的一致性与节点的变动和数据更新过程相关联，全局数据目录本身的一致性则需要采取必要的检测手段和相应的处理措施。类似于基于主副本的定期检查机制[3]并不适合用于分布式并行存储环境，因为节点的对等性使系统没有一个固定的可作为参照的副本节点，而且定期的时间段的确定也是问题，过长达不到一致性检查的目的，过短又将导致大量的消息开销。针对数据目录一致性与数据更新过程的直接相关性，设计了一个数据目录管理和维护的逻辑模块(Data Directory Management and Maintenance, DDMM)。在嵌入数据库执行过程中，由数据库操作请求触发相关数据目录项的一致性检查并对不一致结果进行相应处理，以保证当前事务执行前相关数据目录项的一致性，在数据库执行成功后实时刷新相关数据目录项信息，确保当前事务完成后的数据目录信息一致性。

DPDBS 作为系统平台在宽带视频点播、电子政务等的应用表明，该方案动态解决了数据目录全局信息一致性和实时性问题，并且由于捎带、并行处理等技术的使用，使该模块在低开销的情况下确保了全局数据信息的一致性和实时性，为确保全局数据一致性奠定了基础。

数据目录是在 DPDBS 运行过程中，用于存储数据库分布等信息的一个全局信息表。因此，数据目录与数据字典相比，它属于大粒度库级数据信息，其正确性和一致性既是表级数据信息一致的前提，同时也是系统中各数据库副本保持正确同步的关键。

若系统有n(一般要求$n\geq 3$)个服务器节点，$k(0\leq k<\infty)$个数据库，各数据库分别有$m_l(2\leq l<n)$个副本，则数据目录的结构如图 1 所示。

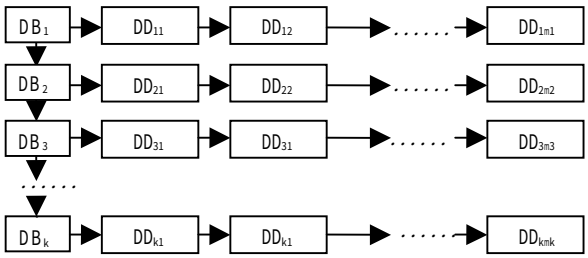

图 1 数据目录结构

其中，$DB_i(0\leq i\leq k)$为数据库名，$DD_{ij}(0\leq i<k, 2\leq j\leq m_l)$为该数据库对应的数据目录项。数据目录项$DD_{ij}$又由数据库所在节点IP地址、数据库副本数、更新版本号等数据项组成，结

作者简介：陈建英(1970—)，女，博士生，主研方向：分布式数据库系统，分布式并行处理；刘心松，教授、博导；谈文蓉，副教授；刘 韬、谭 颖，讲师；王 莉，实验师

收稿日期：2006-02-08 **E-mail：**uestccjy@tom.com

构如表１所示。

表 1 数据目录项结构

NODE_IP	DB_STATE	DB_VERSION	…

数据目录项结构中各项随着数据库所在服务器节点、数据库副本数等的变化而动态改变，各项内容含义如下：

(1)NODE_IP：该数据库所在节点的 IP 地址。

(2)DB_STATE：数据库状态标志信息。

其中，DB_OK_STATE(能够提供数据库服务的正常数据库状态)，DB_RECOVERY_STATE(处于恢复中的数据库状态)，DB_COMPETE_STATE(数据库竞争状态)和 DB_DELE_STATE(正在删除的不可用数据库状态)等。

(3)DB_VESION：数据库最新版本号，即数据目录项被成功更新的最大时间戳，每成功更新一次自动加 1。

(4)…：其余信息项。

2 DDMM 模块的设计与实现

2.1 术语介绍

为了描述 DDMM 算法，先定义如下术语：

(1)主 THD：客户连接某服务器时，由服务器端监听线程创建的用于处理客户数据库操作请求的线程。

(2)代理 THD：主 THD 在拥有当前操作数据库的活动节点上建立的一个数据结构，它与主 THD 有相同的执行参数和环境选项。

(3)主代理 THD：当主 THD 节点不存在当前请求数据库时，由系统指定用于汇集所有代理 THD 的执行结果并决定提交或回滚的一个代理 THD。

2.2 软件总体结构

为了保证数据目录的正确性和一致性，必须采取必要的检测手段和相应的处理措施。因此，数据目录管理和维护逻辑地分为两个子模块：检测子模块和处理子模块。

如图 2 所示，数据目录的管理和维护属于 DPDBS 服务器管理子系统。在节点内部，它从服务器管理子系统中的数据目录管理模块接收和处理数据目录管理信息，必要时调用恢复子系统进行数据库的恢复；在系统中，它借助内部通信子系统交互数据库信息；特别地，该模块与执行子系统密切相关：数据库操作可触发数据目录一致性检查，检查结果不仅参与数据库执行过程，而且在数据库执行完成后刷新数据目录或在检查结果不一致时启动数据目录管理模块进行一致性维护。

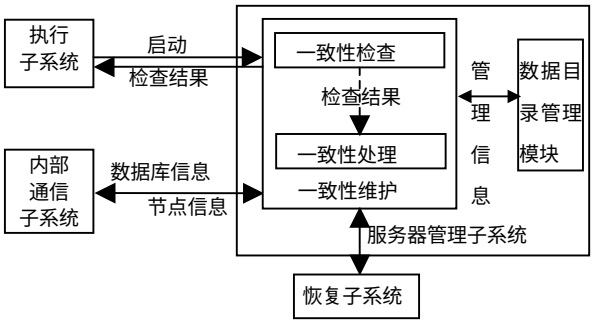

图 2 数据目录管理和维护软件结构

2.3 对 2PC 的改进

DPDBS把每个数据库的操作请求都作为事务处理，事务的执行仍然采用常规的两阶段提交协议 2PC[4~6]，只是针对DPDBS中事务执行的分布性和并行性进行了一些改进，改进的 2PC如图 3 所示。

(1) 图 3 中实线描述主 THD 作为事务发起者时的 2PC 执行过程，虚线描述主代理 THD 作为事务发起者时的 2PC 执行过程。

(2)2PC执行过程：事务发起者发出征求事务提交信息“ Prepare T”，各代理THD收到后在提交表决信息“ Vote ”中稍带本地获取的当前操作库副本数信息S_{LD}(Sum of Local Database)；事务发起者汇总提交表决信息，当且仅当所有节点都同意提交且各节点S_{LD}一致时向各节点发送提交事务信息“ Commit T ”并在收到各代理THD的应答信息“ Ack T ”后正式提交事务；否则发送回滚事务信息“ Abort T ”并在数据目录不一致时启动DDMM维护算法。

(3)一般在正常情况下，事务的最终执行结果由主 THD 返回客户端。

(4)系统中“ 其余节点 ”是与本事务执行无关的节点。

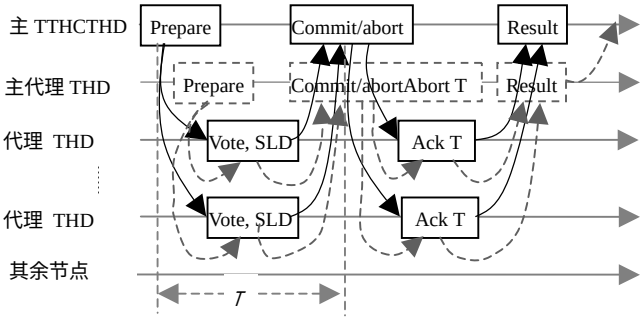

图 3 改进的 2PC

2.4 算法与实现

2.4.1 算法

(1)检测算法

1)服务器端监听到客户对数据库 *DB_a* 的操作请求后在本地创建一个主 THD 并由该主 THD 查询本地数据目录获取 *DB_a* 的 DB_VERSION 及 NODE_IP 等信息。如果 *DB_a* 存在于本地转 2)，否则转 3)。

2)主THD创建各代理THD，如果用户的请求仅为查询，则各代理THD查询本地数据目录获取S_{LD}，如果各节点返回的S_{LD}与本地一致则返回查询结果到客户端后算法结束，否则启动维护算法；如果用户的请求是更新，主THD进行命令分派后启动 2PC协议并等待各代理THD的提交表决信息和S_{LD}，转 4)。

3)主THD创建代理THD并指定主代理THD，如果是查询请求，主THD将用户请求转发到主代理THD，主代理THD收集各代理THD和主THD的S_{LD}信息并与本地进行比较，结果一致则执行数据库查询操作并把查询结果返回主THD，否则启动维护算法；如果是更新请求，主THD分发更新命令到各代理THD和主代理THD后由主代理THD启动 2PC协议并等待各代理THD的表决信息和S_{LD}以及主THD的S_{LD}。

4)将有效时间*Tw*内获取的S_{LD}与本地比较一致且各代理THD都同意提交本地事务时继续 2PC的执行，否则回滚事务并在S_{LD}不一致时启动维护算法。

(2)维护算法

1)主 THD(或主代理 THD)向各代理 THD 发送回滚信息，并指定主 THD 和代理 THD 中 *DB_a* 的副本数最大的节点检查本地数据目录，通知存在 *DB_a* 的所有节点检查本数据库是否存在于本地；

2)被通知节点检查本地数据库，若本库存在则向系统广播该数据库在本节点的信息，转 3)后算法结束；否则广播该数据库不存在，转 4)后算法结束；

3)各节点同步本数据库信息：若在本地数据目录中，则更新其状态等信息；若不存在，则将它加入本地数据目录；

4)检查本地数据目录是否存在本数据库名，若存在则立即删除。

2.4.2 实现

为了对外提供数据库信息，在 DPDBS 中建立了数据目录类结构，它包含数据目录各种操作的一个动态的全局对象，系统中其它模块通过这个全局对象访问数据目录。

(1)数据目录链表结构：存在于系统运行全过程，用于存储数据库分布信息，并随数据库信息的改变而改变。

```
struct db_catalog_item{ // 数据目录项结构
ulong server_ip;    //数据库所在节点 IP
char db_state; //数据库状态
int db_version; //更新版本号
db_catalog_item *next;//数据目录项链表指针
            };
struct db_catalog_list{
 //数据目录链表结构
char db_name; //数据库名
db_catalog_item *item;//指向数据目录项的指针
db_catalog_list *next; //数据目录链表指针
      };
```

(2)数据目录类

```
class db_catalog{
…
int set_db_state(ulong ip, char *db_name, char state);
//数据库存在则更新状态，否则加入该结点
int add_catalog_item(const db_catalog_item * item);
// 在数据目录链表中增加指定数据库信息
int drop_catalog_item(const db_catalog_item * item);
// 在数据目录中删除指定数据库信息
  int drop_catalog_item_of_ip(ulong ip);
//删除数据目录中指定 IP 的所有节点
int get_dbsum_of_ip(ulong ip);
// 获得指定服务器节点的数据库总数
int query_by_ip( ulong ip, db_catalog_list ** da_items );
// 查询指定 IP 的数据库个数
int query_sum_by_db_name( const char * db_name, db_catalog_list ** da_items );// 查询指定数据库个数};
```

2.4.3 存储说明

数据目录在系统启动时就被载入内存且需要常驻内存。

3 结语

根据DPDBS中全局数据目录一致性与数据更新结果的直接相关性，提出了数据目录的动态管理和维护算法DDMM，把数据目录一致性检查嵌入数据库执行过程中，避免了全局数据目录一致性定期检查的不便和由此产生的大量消息开销。实际应用表明，本算法为数据资源的动态的部署和数据冗余度的决策奠定了基础，并与其它相关机制协同实现了数据库系统的高可用性。

参考文献

1 陈建英，刘心松，左朝树等. 基于数据动态冗余的分布式并行系统重构机制[J]. 计算机应用研究, 2004, 21 (11), 229-231.

2 刘心松. 具有分布式并行 I/O 接口的分布式并行服务器系统的性能研究[J]. 电子学报, 2002, 30(12): 1801-1810.

3 魏志强，王先逵，吴　丹等. 异地分布式存储环境下的产品数据一致性控制技术[J]. 计算机集成制造系统, 2003, 9(4): 280-284.

4 Houaily Y A, Chrysanthis P K, Levitan1 S P. An Argument in Favor of the Presumed Commit Protocol[C]. Proc. of 13th Intel. Conf. on Data Engineering, Birmingham England, 1997.

5 Peddemors A J H, Hertzberger L O. A High Performance Distributed Database System for Enhanced Internet Services[J]. FutureGeneration Computer Systems, 1999, 15(3): 407-415.

6 DU C T, WOL P M. Overview of Emerging Database Architectures[J]. Computers and Industrial Engineering, 1997, 32(4): 811.

（上接第 34 页）

它的服务，这些数据服务都被目录服务器发布，并给出了统一的访问端口。

服务请求者只需要访问给定的 Web 端口就会被定向到服务器，在服务器上的服务将被显示出来。

当用户选择合适的数据资源、模型资源等后，平台将自动为用户服务。

在整个过程中，网格操作对用户是透明的，用户不需要知道其资源来自什么地方，只得到网格服务产生的结果。

一个客户端显示的可视化服务结果如图 8 所示。

图 8 客户端界面

4 结论与展望

本文以地形可视化服务为例，设计了地形数据可视化服务的体系框架和功能结构，研究服务运行机制、数据资源代理和服务管理等关键技术；同时也探讨了网格环境下的客户端程序设计和地形可视化方法；最后实现一个应用案例。与网络可视化相比，地形服务方式所带来的好处是显而易见的，它能够利用网格技术，自动收集、查询、处理、分析各种地理分散、平台异构的空间数据，通过动态的网格服务，更为方便快捷地实现数据共享。在此基础上，可以开展其它地形可视化应用服务研究，如辅助流域坝系规划与设计，遥感专题信息分析等。

参考文献

1 Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations[EB/OL]. http://www.globus.org/alliance/publications/papers/anatomy.pdf, 2001

2 Foster I, Kesselman C. The Grid: Blueprint for a New Computing Infrastructure[M]. San Fransisco: Morgan Kaufmann, 1999.

3 胡　敏，顾君忠. Globus 网络体系结构及其服务的实现[J]. 计算机工程, 2003, 29(15):5-7.

4 Global Grid Forum Homepage[EB/OL]. http://www.globalgridforum.com/, 1999.

5 都志辉，陈　渝，刘　鹏等. 以服务为中心的网格体系结构 OGSA[J]. 计算机科学, 2003, 30(7): 26-29.

6 Foster I, Kesselman C, Nick J M, et al. Grid Services for Distributed Integration[J]. Computer, 2002, 35(6): 37-46.

7 朱　军. 基于 VRML 的大型地形环境建模研究[D]. 重庆: 西南交通大学, 2003.

8 王意洁，肖　侬，任　浩等. 数据网格及其关键技术研究[J]. 计算机研究与发展, 2002, 39(8): 943-947.

9 朱　军，龚建华. 大规模地形实时绘制算法[J]. 地理与地理信息科学, 2005, 21(2): 24-27.

10 李　妮，彭晓源，刘　杰. 计算网格及其在虚拟样机协同环境中的用探讨[J]. 系统仿真学报, 2004, 16(2): 247-250.